

IN THE CLAIMS:

THIS LISTING OF CLAIMS WILL REPLACE ALL PRIOR VERSIONS, AND LISTINGS, OF CLAIMS IN THE APPLICATION.

LISTING OF CLAIMS:

1. (currently amended) In a computer network having a plurality of nodes for interacting with computer network information, a system for managing said plurality of nodes comprising:

means for establishing a DDB in each of said nodes; and,

means for controlling contents of each said DDB to be ~~substantially~~ identical to contents of every other said DDB and in a manner to avoid a single point of failure.

2. (original) The system of claim 1 and wherein:

said computer network information comprises both computer data and domain configuration status, and said each of said nodes has a unique IP address; and,

said DDB establishing means further comprises:

means for associating each said unique IP address with its respective node to provide an IP-address-respective-node association;

means for combining said association for said each of said nodes into a network IP association; and,

means for distributing said network IP association to said DDB in each of said nodes; and,

said contents controlling means further comprises:

means for maintaining the most current of said domain configuration
status in said DDB in each of said nodes.

3. (original) The system of claim 1 and wherein said interacting includes receiving,
storing, modifying, and transmitting.

4. (currently amended) The system of claim 2 and wherein said controlling means
further comprises:

means for selecting one of said plurality of nodes as a master node;

means for subordinating all other of said plurality of nodes to said master node in
a configuration defined by said master node and said all other of said plurality of nodes;
and,

said master node including means for responding to a change to said domain
configuration status in a manner to maintain said contents of each said DDB ~~substantially~~
identical to said contents of every other DDB.

5. (original) The system of claim 4 and wherein said controlling means further
comprises:

means for replacing said master node with another node if said master node fails.

6. (original) The system of claim 5 and wherein said master node replacing means
includes means for replacing said master node with another node selected from said
configuration.

7. (original) The system of claim 4 and wherein said change to said domain configuration status is selected from the group of changes consisting of: adding a first node to said configuration; deleting a second node from said configuration; a third node failing in said configuration; and, a network link failing between a fourth node in said configuration and said master node.

8. (original) The system of claim 7 and wherein said computer network is a client-server network having a graphical user interface and wherein said selecting means further comprises:

means, utilizing said graphical user interface, for invoking a select master dialog by which said user can select said one of said plurality of nodes.

9. (original) The system of claim 5 and wherein said computer network is a client-server network having a graphical user interface and wherein said replacing means further comprises:

means, utilizing said graphical user interface, for invoking a select master dialog by which said user can select said another node.

10. (original) The system of claim 7 and wherein said responding means further comprises:

first means for handling said third node failing under conditions in which said master node is known to said third node; and,

second means for handling said third node failing under conditions in which said master node is unknown to said third node.

11. (original) The system of claim 10 and wherein said first means further comprises:

means for establishing version numbers to identify versions of said DDB in each of said plurality of nodes;

means, employed by each of said all other of said plurality of nodes, for continuously polling said master node at regular intervals to obtain the most current one of said version numbers of the DDB in said master node;

means employed by said master node, responsive to said polling received from said third node after recovery of said third node, for sending said most current one of said version numbers to said third node; and

means for updating the DDB in said third node if said most current one of said version numbers does not match the version number of said DDB in said third node.

12. (original) The system of claim 11 and wherein said third node DDB updating means includes means for handshaking between said master node and said third node.

13. (original) The system of claim 10 and wherein said conditions in which said master node is unknown to said third node include both said third node failing while it is being added to said configuration and said master node was replaced during time of failure of said third node.

14. (original) The system of claim 13 and wherein said second means further comprises:

means, employed by said master node, for repetitively pinging said third node at predetermined intervals until said third node recovers and sends a recovery signal to said master node; and,

said master node including means, responsive to said recovery signal, for updating the DDB in said third node as may be needed.

15. (original) The system of claim 14 and wherein said third node DDB updating means includes means for handshaking between said master node and said third node.

16. (original) The system of claim 7 and wherein said responding means further comprises:

means for handling said network link failing.

17. (original) The system of claim 16 and wherein said network link failing handling means further comprises:

means for establishing version numbers to identify versions of said DDB in each of said plurality of nodes;

means, employed by each of said all other of said plurality of nodes, for continuously polling said master node at regular intervals to obtain the most current one of said version numbers of the DDB in said master node;

means employed by said master node, responsive to said polling received from said fourth node after recovery of said network link, for sending said most current one of said version numbers to said fourth node; and,

means for updating the DDB in said fourth node if said most current one of said version numbers does not match the version number of said DDB in said fourth node.

18. (original) The system of claim 17 and wherein said fourth node DDB updating means includes means for handshaking between said master node and said fourth node.

19. (original) The system of claim 7 and wherein said responding means further comprises:

means for handling said adding a first node to said configuration.

20. (original) The system of claim 19 and wherein said first node adding handling means comprises:

means for determining if said first node is being added through said master node to obtain a master-added node or through one of said all other of said plurality of nodes to obtain a portal-added node.

21. (original) The system of claim 20 and wherein said determining means, for the condition of said master-added node, further comprises:

said master node including means for updating the DDB in said master node with the IP address of said first node and for informing said first node that the first node's master is said master node;

said first node including means, responsive to operation of said informing means, for entering the IP address of said master node in the DDB of said first node and for acknowledging said master node; and,

said master node including means for sending said IP address of said first node as an update to all other nodes in said configuration.

22. (original) The system of claim 21 and wherein said IP address sending means further comprises:

means for performing a master to node handshake between said master node and said all other nodes in said configuration.

23. (original) The system of claim 20 and wherein said determining means, for the condition of said portal-added node, further comprises:

a cache memory included in said portal node;

means for holding the IP address of said first node in said cache memory;

means for performing a node to master handshake between said first node and said master node;

said portal node including means for informing said master node of the IP address of said first node;

said master node including means for updating the DDB in said master node with the IP address of said first node and for informing said first node that the first node's master is said master node;

said first node including means, responsive to operation of said informing means, for entering the IP address of said master node in the DDB of said first node and for acknowledging said master node; and,

said master node including means for sending said IP address of said first node as an update to all other nodes in said configuration.

24. (original) The system of claim 23 and wherein said IP address sending means further comprises:

means for performing a master to node handshake between said master node and said all other nodes in said configuration.

25. (original) The system of claim 8 and wherein said responding means further comprises:

means for handling said deleting a second node from said plurality of nodes.

26. (original) The system of claim 25 and wherein said second node deleting handling means further comprises:

means, utilizing said graphical user interface, for removing said second node from said configuration;

means for determining if said second node is removed through said master node;

means, responsive to operation of said determining means removing said second node through said master node, for:

- (1) updating the DDB in said master node by removing the IP address of said second node from the DDB of said master node;
- (2) informing said second node that said configuration no longer includes said second node and detaching said second node from said configuration;
- (3) erasing all contents of the DDB of said second node; and,
- (4) sending an update to all remaining nodes in said configuration.

27. (original) The system of claim 26 and wherein said update sending means further comprises:

means for performing a master to node handshake between said master node and said all remaining nodes in said configuration.

28. (original) The system of claim 25 and wherein said second node deleting handling means further comprises:

means, utilizing said graphical user interface, for removing said second node from said configuration;

means for selecting a portal-removal node other than said master node through which to remove said second node from said configuration;

a cache memory included in said portal-removal node;

means for determining if said second node is removed through said master node;

means, responsive to operation of said determining means removing said second node through said portal-removal node and not through said master node, for:

- (1) storing the IP address of said second node in said cache;
- (2) performing a node to master handshake between said portal-removal node and said master node;
- (3) informing said master node to remove the IP address of said second node from the DDB of said master node;
- (4) updating the DDB in said master node by removing the IP address of said second node from the DDB of said master node;
- (5) informing said second node that said configuration no longer includes said second node and detaching said second node from said configuration;
- (6) erasing all contents of the DDB of said second node; and,
- (7) sending an update to all remaining nodes in said configuration.

29. (original) The system of claim 28 and wherein said update sending means further comprises:

means for performing a master to node handshake between said master node and said all remaining nodes in said configuration.

30. (original) The system of claim 22 or 24 and wherein said master to node handshake performing means comprises for each one of said all other nodes in said configuration:

first deciding means for deciding if the IP address of said master node in said update matches the IP address of said master node contained in the DDB of said each one of said all other nodes in said configuration;

means, responsive to operation of said first deciding means deciding no match, for rejecting said update and logging said event;

second deciding means, responsive to operation of said first deciding means deciding a match, for deciding if the version number of the DDB in said master node before said update matches the version number of said DDB of said each one of said all other nodes in said configuration before said update;

means, responsive to operation of said second deciding means deciding no match, for accepting a completely updated DDB with updated version number from said master node; and,

means, responsive to operation of said second deciding means deciding a match, for accepting only said update with said updated version number from said master node.

31. (original) The system of claim 27 or 29 and wherein said master to node handshake performing means comprises for each one of said all remaining nodes in said configuration:

first deciding means for deciding if the IP address of said master node in said update matches the IP address of said master node contained in the DDB of said each one of said all remaining nodes in said configuration;

means, responsive to operation of said first deciding means deciding no match, for rejecting said update and logging said event;

second deciding means, responsive to operation of said first deciding means deciding a match, for deciding if the version number of the DDB in said master node before said update matches the version number of said DDB of said each one of said all remaining nodes in said configuration;

means, responsive to operation of said second deciding means deciding no match, for accepting a completely updated DDB with updated version number from said master node; and,

means, responsive to operation of said second deciding means deciding a match, for accepting only said update with said updated version number from said master node.

32. (currently amended) A computer program product for use in a computer network having a plurality of nodes for interacting with computer network information, said computer program product including a computer usable medium having computer readable program code thereon for managing said plurality of nodes, said program code comprising:

program code for establishing a DDB in each of said nodes; and,

program code for controlling contents of each said DDB to be ~~substantially~~ identical to contents of every other said DDB and in manner to avoid a single point of failure.

33. (original) The computer program product of claim 32 and wherein:

said computer network information comprises both computer data and domain configuration status, and said each of said nodes has a unique IP address; and,

said DDB establishing program code further comprises:

program code for associating each said unique IP address with its
respective node to provide an IP-address-respective-node association;
program code for combining said association for said each of said nodes
into a network IP association; and,
program code for distributing said network IP association to said DDB in
each of said nodes; and,

said contents controlling program code further comprises:

program code for maintaining the most current of said domain
configuration status in said DDB in each of said nodes.

34. (original) The computer program product of claim 32 and wherein said interacting includes receiving, storing, modifying, and transmitting.

35. (currently amended) The computer program product of claim 33 and wherein said controlling program code further comprises:

program code for selecting one of said plurality of nodes as a master node;
program code for subordinating all other of said plurality of nodes to said master node in a configuration defined by said master node and said all other of said plurality of nodes; and,

said master node including program code for responding to a change to said domain configuration status in a manner to maintain said contents of each said DDB ~~substantially~~ identical to said contents of every other DDB.

36. (original) The computer program product of claim 35 and wherein said controlling program code further comprises:

program code for replacing said master node with another node if said master node fails.

37. (original) The computer program product of claim 36 and wherein said master node replacing program code includes program code for replacing said master node with another node selected from said configuration.

38. (original) The computer program product of claim 35 and wherein said change to said domain configuration status is selected from the group of changes consisting of: adding a first node to said configuration; deleting a second node from said configuration; a third node failing in said configuration; and, a network link failing between a fourth node in said configuration and said master node.

39. (original) The computer program product of claim 38 and wherein said computer network is a client-server network having a graphical user interface and wherein said selecting program code further comprises:

program code, utilizing said graphical user interface, for invoking a select master dialog by which said user can select said one of said plurality of nodes.

40. (original) The computer program product of claim 36 and wherein said computer network is a client-server network having a graphical user interface and wherein said replacing program code further comprises:

program code, utilizing said graphical user interface, for invoking a select master dialog by which said user can select said another node.

41. (original) The computer program product of claim 38 and wherein said responding program code further comprises:

first program code for handling said third node failing under conditions in which said master node is known to said third node; and,

second program code for handling said third node failing under conditions in which said master node is unknown to said third node.

42. (original) The computer program product of claim 41 and wherein said first program code further comprises:

program code for establishing version numbers to identify versions of said DDB in each of said plurality of nodes;

program code, employed by each of said all other of said plurality of nodes, for continuously polling said master node at regular intervals to obtain the most current one of said version numbers of the DDB in said master node;

program code employed by said master node, responsive to said polling received from said third node after recovery of said third node, for sending said most current one of said version numbers to said third node; and

program code for updating the DDB in said third node if said most current one of said version numbers does not match the version number of said DDB in said third node.

43. (original) The computer program product of claim 42 and wherein said third node DDB updating program code includes program code for handshaking between said master node and said third node.

44. (original) The computer program product of claim 41 and wherein said conditions in which said master node is unknown to said third node include both said third node failing while it is being added to said configuration and said master node was replaced during time of failure of said third node.

45. (original) The computer program product of claim 44 and wherein said second program code further comprises:

program code, employed by said master node, for repetitively pinging said third node at predetermined intervals until said third node recovers and sends a recovery signal to said master node; and,

said master node including program code, responsive to said recovery signal, for updating the DDB in said third node as may be needed.

46. (original) The computer program product of claim 45 and wherein said third node DDB updating program code includes program code for handshaking between said master node and said third node.

47. (original) The computer program product of claim 38 and wherein said responding program code further comprises:

program code for handling said network link failing.

48. (original) The computer program product of claim 47 and wherein said network link failing handling program code further comprises:

program code for establishing version numbers to identify versions of said DDB in each of said plurality of nodes;

program code, employed by each of said all other of said plurality of nodes, for continuously polling said master node at regular intervals to obtain the most current one of said version numbers of the DDB in said master node;

program code employed by said master node, responsive to said polling received from said fourth node after recovery of said network link, for sending said most current one of said version numbers to said fourth node; and,

program code for updating the DDB in said fourth node if said most current one of said version numbers does not match the version number of said DDB in said fourth node.

49. (original) The computer program product of claim 48 and wherein said fourth node DDB updating program code includes program code for handshaking between said master node and said fourth node.

50. (original) The computer program product of claim 38 and wherein said responding program code further comprises:

program code for handling said adding a first node to said configuration.

51. (original) The computer program product of claim 50 and wherein said first node adding handling program code comprises:

program code for determining if said first node is being added through said master node to obtain a master-added node or through one of said all other of said plurality of nodes to obtain a portal-added node.

52. (original) The computer program product of claim 51 and wherein said determining program code, for the condition of said master-added node, further comprises:

said master node including program code for updating the DDB in said master node with the IP address of said first node and for informing said first node that the first node's master is said master node;

said first node including program code, responsive to operation of said informing program code, for entering the IP address of said master node in the DDB of said first node and for acknowledging said master node; and,

said master node including program code for sending said IP address of said first node as an update to all other nodes in said configuration.

53. (original) The computer program product of claim 52 and wherein said IP address sending program code further comprises:

program code for performing a master to node handshake between said master node and said all other nodes in said configuration.

54. (original) The computer program product of claim 51 and wherein said determining program code, for the condition of said portal-added node, further comprises:

a cache memory included in said portal node;

program code for holding the IP address of said first node in said cache memory;

program code for performing a node to master handshake between said first node and said master node;

said portal node including program code for informing said master node of the IP address of said first node;

said master node including program code for updating the DDB in said master node with the IP address of said first node and for informing said first node that the first node's master is said master node;

said first node including program code, responsive to operation of said informing program code, for entering the IP address of said master node in the DDB of said first node and for acknowledging said master node; and,

said master node including program code for sending said IP address of said first node as an update to all other nodes in said configuration.

55. (original) The computer program product of claim 54 and wherein said IP address sending program code further comprises:

program code for performing a master to node handshake between said master node and said all other nodes in said configuration.

56. (original) The computer program product of claim 39 and wherein said responding program code further comprises:

program code for handling said deleting a second node from said plurality of nodes.

57. (original) The computer program product of claim 56 and wherein said second node deleting handling program code further comprises:

program code, utilizing said graphical user interface, for removing said second node from said configuration;

program code for determining if said second node is removed through said master node;

program code, responsive to operation of said determining program code removing said second node through said master node, for:

(1) updating the DDB in said master node by removing the IP address of said second node from the DDB of said master node;

- (2) informing said second node that said configuration no longer includes said second node and detaching said second node from said configuration;
- (3) erasing all contents of the DDB of said second node; and,
- (4) sending an update to all remaining nodes in said configuration.

58. (original) The computer program product of claim 57 and wherein said update sending program code further comprises:

program code for performing a master to node handshake between said master node and said all remaining nodes in said configuration.

59. (original) The computer program product of claim 56 and wherein said second node deleting handling program code further comprises:

program code, utilizing said graphical user interface, for removing said second node from said configuration;

program code for selecting a portal-removal node other than said master node through which to remove said second node from said configuration;

a cache memory included in said portal-removal node;

program code for determining if said second node is removed through said master node;

program code, responsive to operation of said determining program code removing said second node through said portal-removal node and not through said master node, for:

- (1) storing the IP address of said second node in said cache;

(2) performing a node to master handshake between said portal-removal node and said master node;

(3) informing said master node to remove the IP address of said second node from the DDB of said master node;

(4) updating the DDB in said master node by removing the IP address of said second node from the DDB of said master node;

(5) informing said second node that said configuration no longer includes said second node and detaching said second node from said configuration;

(6) erasing all contents of the DDB of said second node; and,

(7) sending an update to all remaining nodes in said configuration.

60. (original) The computer program product of claim 59 and wherein said update sending program code further comprises:

program code for performing a master to node handshake between said master node and said all remaining nodes in said configuration.

61. (original) The computer program product of claim 53 or 55 and wherein said master to node handshake performing program code comprises for each one of said all other nodes in said configuration:

first deciding program code for deciding if the IP address of said master node in said update matches the IP address of said master node contained in the DDB of said each one of said all other nodes in said configuration;

program code, responsive to operation of said first deciding program code
deciding no match, for rejecting said update and logging said event;

second deciding program code, responsive to operation of said first deciding
program code deciding a match, for deciding if the version number of the DDB in said
master node before said update matches the version number of said DDB of said each one
of said all other nodes in said configuration before said update;

program code, responsive to operation of said second deciding program code
deciding no match, for accepting a completely updated DDB with updated version
number from said master node; and,

program code, responsive to operation of said second deciding program code
deciding a match, for accepting only said update with said updated version number from
said master node.

62. (original) The computer program product of claim 58 or 60 and wherein said
master to node handshake performing program code comprises for each one of said all
remaining nodes in said configuration:

first deciding program code for deciding if the IP address of said master node in
said update matches the IP address of said master node contained in the DDB of said each
one of said all remaining nodes in said configuration;

program code, responsive to operation of said first deciding program code
deciding no match, for rejecting said update and logging said event;

second deciding program code, responsive to operation of said first deciding
program code deciding a match, for deciding if the version number of the DDB in said

master node before said update matches the version number of said DDB of said each one of said all remaining nodes in said configuration;

program code, responsive to operation of said second deciding program code deciding no match, for accepting a completely updated DDB with updated version number from said master node; and,

program code, responsive to operation of said second deciding program code deciding a match, for accepting only said update with said updated version number from said master node.

63. (currently amended) In a computer network having a plurality of nodes for interacting with computer network information, a method for managing said plurality of nodes comprising:

establishing a DDB in each of said nodes; and,
controlling contents of each said DDB to be ~~substantially~~ identical to contents of every other said DDB and in a manner to avoid a single point of failure.

64. (original) The method of claim 63 and wherein:

said computer network information comprises both computer data and domain configuration status, and said each of said nodes has a unique IP address; and,
said DDB establishing further comprises:

associating each said unique IP address with its respective node to provide
an IP-address-respective-node association;

combining said association for said each of said nodes into a network IP
association; and,

distributing said network IP association to said DDB in each of said nodes;
and,

said contents controlling further comprises:

maintaining the most current said domain configuration status in said
DDB in each of said nodes.

65. (original) The method of claim 63 and wherein said interacting includes receiving,
storing, modifying, and transmitting.

66. (currently amended) The method of claim 64 and wherein said controlling further
comprises:

selecting one of said plurality of nodes as a master node;
subordinating all other of said plurality of nodes to said master node in a
configuration defined by said master node and said all other of said plurality of nodes;
and,

responding to a change to said domain configuration status in a manner to
maintain said contents of each said DDB ~~substantially~~ identical to said contents of every
other DDB.

67. (original) The method of claim 66 and wherein said controlling further comprises:
replacing said master node with another node if said master node fails.

68. (original) The method of claim 67 and wherein said master node replacing
includes replacing said master node with another node selected from said configuration.

69. (original) The method of claim 66 and wherein said change to said domain
configuration status is selected from the group of changes consisting of: adding a first
node to said configuration; deleting a second node from said configuration; a third node
failing in said configuration; and, a network link failing between a fourth node in said
configuration and said master node.

70. (original) The method of claim 69 and wherein said computer network is a client-
server network having a graphical user interface and wherein said selecting further
comprises:

utilizing said graphical user interface to invoke a select master dialog by which
said user can select said one of said plurality of nodes.

71. (original) The method of claim 67 and wherein said computer network is a client-
server network having a graphical user interface and wherein said replacing further
comprises:

utilizing said graphical user interface to invoke a select master dialog by which
said user can select said another node.

72. (original) The method of claim 69 and wherein said responding further comprises:
first handling said third node failing under conditions in which said master node is known to said third node; and,

second handling said third node failing under conditions in which said master node is unknown to said third node.

73. (original) The method of claim 72 and wherein said first handling further comprises:

establishing version numbers to identify versions of said DDB in each of said plurality of nodes;

each of said all other of said plurality of nodes continuously polling said master node at regular intervals to obtain the most current one of said version numbers of the DDB in said master node;

said master node, responsive to said polling received from said third node after recovery of said third node, sending said most current one of said version numbers to said third node; and

updating the DDB in said third node if said most current one of said version numbers does not match the version number of said DDB in said third node.

74. (original) The method of claim 73 and wherein said third node DDB updating includes handshaking between said master node and said third node.

75. (original) The method of claim 72 and wherein said conditions under which said master node is unknown to said third node include both said third node failing while being added to said configuration and said master node being replaced during time of failure of said third node.

76. (original) The method of claim 75 and wherein said second handling further comprises:

said master node repetitively pinging said third node at predetermined intervals until said third node recovers and sends a recovery signal to said master node; and,

said master node, responsive to said recovery signal, updating the DDB in said third node as may be needed.

77. (original) The method of claim 76 and wherein said third node DDB updating includes handshaking between said master node and said third node.

78. (original) The method of claim 69 and wherein said responding further comprises:
handling said network link failing.

79. (original) The method of claim 78 and wherein said network link failing handling further comprises:

establishing version numbers to identify versions of said DDB in each of said plurality of nodes;

each of said all other of said plurality of nodes continuously polling said master node at regular intervals to obtain the most current one of said version numbers of the DDB in said master node;

said master node, responsive to said polling received from said fourth node after recovery of said network link, sending said most current one of said version numbers to said fourth node; and,

updating the DDB in said fourth node if said most current one of said version numbers does not match the version number of said DDB in said fourth node.

80. (original) The method of claim 79 and wherein said fourth node DDB updating includes handshaking between said master node and said fourth node.

81. (original) The method of claim 69 and wherein said responding further comprises: handling said adding a first node to said configuration.

82. (original) The method of claim 81 and wherein said first node adding handling comprises:

determining if said first node is being added through said master node to obtain a master-added node or through one of said all other of said plurality of nodes to obtain a portal-added node.

83. (original) The method of claim 82 and wherein said determining, for the condition of said master-added node, further comprises:

said master node updating the DDB in said master node with the IP address of said first node and informing said first node that the first node's master is said master node;

said first node, responsive to said informing, entering the IP address of said master node in the DDB of said first node and acknowledging said master node; and,

said master node sending said IP address of said first node as an update to all other nodes in said configuration.

84. (original) The method of claim 83 and wherein said IP address sending further comprises:

performing a master to node handshake between said master node and said all other nodes in said configuration.

85. (original) The method of claim 82 for the condition of said portal-added node, and wherein said portal-added node includes a cache memory, said determining further comprising:

holding the IP address of said first node in said cache memory;

performing a node to master handshake between said first node and said master node;

said portal node informing said master node of the IP address of said first node;

said master node updating the DDB in said master node with the IP address of said first node and informing said first node that the first node's master is said master node;

said first node, responsive to said informing, entering the IP address of said master node in the DDB of said first node and acknowledging said master node; and, said master node sending said IP address of said first node as an update to all other nodes in said configuration.

86. (original) The method of claim 85 and wherein said IP address sending further comprises:

performing a master to node handshake between said master node and said all other nodes in said configuration.

87. (original) The method of claim 70 and wherein said responding further comprises: handling said deleting a second node from said plurality of nodes.

88. (original) The method of claim 87 and wherein said second node deleting handling further comprises:

utilizing said graphical user interface to remove said second node from said configuration;

determining if said second node is removed through said master node;
said determining, responsive to said utilizing removing said second node through said master node, for:

(1) updating the DDB in said master node by removing the IP address of said second node from the DDB of said master node;

- (2) informing said second node that said configuration no longer includes said second node and detaching said second node from said configuration;
 - (3) erasing all contents of the DDB of said second node; and,
 - (4) sending an update to all remaining nodes in said configuration.
89. (original) The method of claim 88 and wherein said update sending further comprises:
- performing a master to node handshake between said master node and said all remaining nodes in said configuration.
90. (original) The method of claim 87 and wherein said second node deleting handling further comprises:
- utilizing said graphical user interface to remove said second node from said configuration;
 - selecting a portal-removal node other than said master node through which to remove said second node from said configuration;
 - establishing a cache memory in said portal-removal node;
 - determining if said second node is removed through said master node;
 - said determining, responsive to said utilizing removing said second node through said portal-removal node and not through said master node:
 - (1) storing the IP address of said second node in said cache;
 - (2) performing a node to master handshake between said portal-removal node and said master node;

(3) informing said master node to remove the IP address of said second node from the DDB of said master node;

(4) updating the DDB in said master node by removing the IP address of said second node from the DDB of said master node;

(5) informing said second node that said configuration no longer includes said second node and detaching said second node from said configuration;

(6) erasing all contents of the DDB of said second node; and,

(7) sending an update to all remaining nodes in said configuration.

91. (original) The method of claim 90 and wherein said update sending further comprises:

performing a master to node handshake between said master node and said all remaining nodes in said configuration.

92. (original) The method of claim 84, 86, 89, or 91 and wherein said master to node handshake performing comprises for each one of said all other nodes in said configuration:

deciding if the IP address of said master node in said update matches the IP address of said master node contained in the DDB of said each one of said all other nodes in said configuration;

if no IP address match, rejecting said update and logging said event;

if an IP address match, deciding if the version number of the DDB in said master node before said update matches the version number in the DDB of said each one of said all other nodes in said configuration before said update;

if no version number match, accepting a completely updated DDB with updated version number from said master node; and,

if a version number match, accepting only said update with said updated version number from said master node.